

--Preliminary--  
Science Data Access Architectures  
Mike Martin, 11/20/06

This paper presents a description and comparison of several architectures for providing access to distributed science data collections. The architectures include the Open GIS Consortium's Open Web Services (OWS), the International Virtual Observatory Alliance (IVOA), the Planetary Data System's implementation of Object Oriented Data Technology (PDS-ODT) and the Earth Observing System Clearinghouse (ECHO). The OWS provides specifications for accessing digital maps and images, geographic information system (GIS) data and services. The IVOA provides specifications for accessing astrophysics registries, catalogs, data and services. The PDS-ODT system provides servers to access distributed planetary science registries, catalogs and data. The ECHO system provides a central catalog and order system for accessing distributed collections of earth science data and processing services. These are all "bolt-on" architectures that are added to existing data repositories as middleware to provide standard access protocols. There are many other access architectures in use or under development but it is felt that these four represent a good range of approaches to contemporary distributed data access.

The goals of this paper are:

- to describe how each scheme works
- to compare the different approaches to common problems
- to identify any exemplary or noteworthy features of the various schemes

## **Terminology**

The descriptions will focus on the services that each architecture provides. The "Service Oriented Architecture" (SOA) has become a prominent buzzword describing an architecture with self-describing, discoverable, loosely-coupled, platform-independent, modular components. A main thrust is to commoditize information and computer resources for automated consumption. The term "Web Services" has come to have a special meaning to some people as an SOA which utilizes a Web Service Description Language (WSDL) XML text file to describe a service, uses Simple Object Access Protocol (SOAP) XML text messages for service requests and responses and may also use a Universal Description, Discovery and Integration (UDDI) registry to store and access the service descriptions. There is also a whole suite of "WS-" specifications that provide integrated Security, Reliable Messaging (WSRM), Resource Framework (WSRF), Distributed Management (WSDM) and Notification. To some this is utopia, to others an anathema. All of the architectures described in this paper are SOA's to some extent. A couple are on the forefront in adopting elements of the "Web Services" architecture. Within the paper I will highlight this special meaning of "Web Services" by quoting it. The term collection is used for high level metadata which describes a collection or dataset and the term catalog is used for metadata which describes individual products or granules.

## **Open Web Services (OWS)**

One of the early implementations of distributed access to geographic data collections was the Web Mapping Testbed sponsored by the Open GIS Consortium and NASA's Digital Earth program with many other government and commercial partners. The goal was to provide uniform access to hundreds

of geographic data collections sitting at government, educational and commercial sites. These collections were generally maintained in a variety of internal storage formats with custom retrieval and display software.

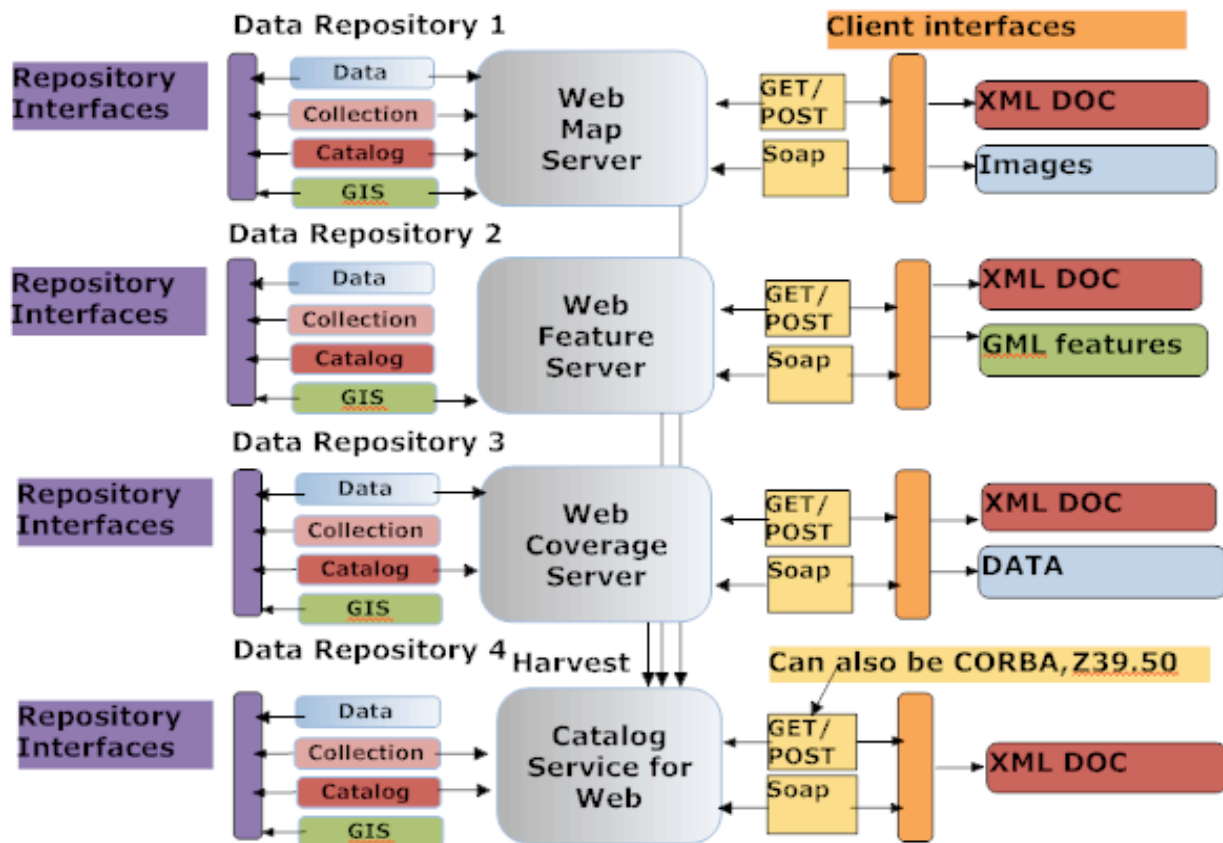


Figure 1 – OGC Web Services

Description. The Open Geospatial Consortium, Inc. (OGC) determines a need for and develops specifications for services. Figure 1 illustrates the major components of the Open Web Service (OWS). The data provider implements a web service (based on the OGC specifications) to provide access to a data collection. The web map service (WMS) will provide renditions of the data collection as simple images. The client sends a GetCapabilities request to a server to identify the spatial area, layers (types of data), image formats (JPEG, PNG or GIF) and projections that are available. After determining which layers are needed the client sends a GetMap request indicating the area of interest, layers, formats and projections that are desired and the server produces and transmits the images. The web feature service (WFS) responds to a GetFeature request to access a geographic database and return feature information such as lines or polygons representing roads, rivers or boundaries represented in Geography Markup Language (GML). The web coverage server (WCS) will respond to a GetCoverage request by providing the original source data (e.g. multispectral images) in a scientific data format (GEOTIFF, HDF) suitable for further analysis. The web coverage server may also provide detailed metadata (described using XML Schema) for the data collection or for individual products. A catalog server for web (CSW) builds an inventory of available servers by harvesting a minimal set of descriptive information (content, spatial, temporal) from all the servers it is aware of. It may also be loaded through a transaction process with additional metadata conforming to a standard like ISO-19115, Geographic Information - Metadata.

The client queries the CSW with a GetRecords request to identify OWS servers containing data collections of interest or to extract detailed metadata (again described using XML Schema). There are two related specifications, Styled Layer Descriptors which can be used to modify the display of feature data in a map display (change the color of a road, for example) and Web Map Context to capture a representation of different map layers from different servers as an entity that can be saved, edited and reused. There are numerous other services being developed to provide gazetteer, data processing, coordinate transformation, terrain and image classification services.

Service summary. The catalog service for web provides a registry of web services and a metadata catalog of their contents. The catalog service provides the following operations:

GetCapabilities	- describes the service, lists operations, metadata types and filter operators.
DescribeRecords	- presents a schema for searchable metadata types.
GetRecords	- performs a metadata search.
GetRecordsById	- retrieves individual records based on the results of a GetRecords operation.
GetDomain	- provides the value range for a searchable property or request parameter.
Harvest	- collects metadata from a web service (WMS, WCS, WFS).
Transaction	- provides a mechanism to insert, update or delete catalog records.

There are also several other operations defined for compatibility with existing servers which use the Z39.50 or CORBA protocols. They include session related operations "Initialize", "Close", "Status" and "Cancel" and the brokered access operation "Order".

The web map service provides the following operations:

GetCapabilities	- describes what "layers" of data are available and other display options.
GetMap	- select layers for display, output image size, format and projection, transparency.
GetFeatureInfo	- identify features associated with a certain x,y position in specified map layers.

The web feature service provides the following operations:

GetCapabilities	- indicates the descriptive formats that are available for the DescribeFeatureType and GetFeature operations, and lists available feature types and query capabilities.
DescribeFeatureType	- provides a schema definition for the specified feature type.
GetFeature	- returns a GML or other rendition of the feature data.

The web coverage service provides access to detailed metadata and source data for the rendered images presented by the WMS.

GetCapabilities	- identifies the coverages that are available on a server.
DescribeCoverage	- provides detailed metadata and coverage format delivery options.
GetCoverage	- delivers the metadata or coverage data in the specified format.

The following examples show how to use the GetCapabilities and GetMap services from JPL's OnEarth server using the HTTP GET protocol.

<http://wms.jpl.nasa.gov/wms.cgi?version=1.1.1&service=WMS&request=GetCapabilities>

[http://wms.jpl.nasa.gov/wms.cgi?request=GetMap&layers=global\\_mosaic&srs=EPSG:4326&format=image/jpeg&styles=visual&bbox=-180,-90,180,90&width=800&height=400](http://wms.jpl.nasa.gov/wms.cgi?request=GetMap&layers=global_mosaic&srs=EPSG:4326&format=image/jpeg&styles=visual&bbox=-180,-90,180,90&width=800&height=400)

Summary. The Open Web Services provide an elegant model for discovery and access to geographic information resources. Unfortunately, the vision has not yet been fully realized and it is fairly difficult to locate resources at this time. The open source server and client implementations do not seem to be very robust. This architecture combines the service registry with the catalog metadata search service which adds complexity. It would seem simpler to have a catalog service closely associated with the map, feature and coverage services and have a registry service as a separate entity. There is no standard mechanism to register one's servers with a higher authority.

### International Virtual Observatory Alliance (IVOA)

The goal of the Virtual Observatory is to improve and unify access to astronomical data and services. As the web map server has allowed enhanced access to geographic data, the cone search protocol (search on right ascension, declination and search radius) along with the Virtual Observatory Table "VOTable" XML response format have allowed easy access to an enormous collection of astronomical data. Not only is it easy to retrieve information from a specific server, but retrieval engines like DataScope can search dozens of servers and retrieve, integrate and present hundreds of resources in a matter of minutes.

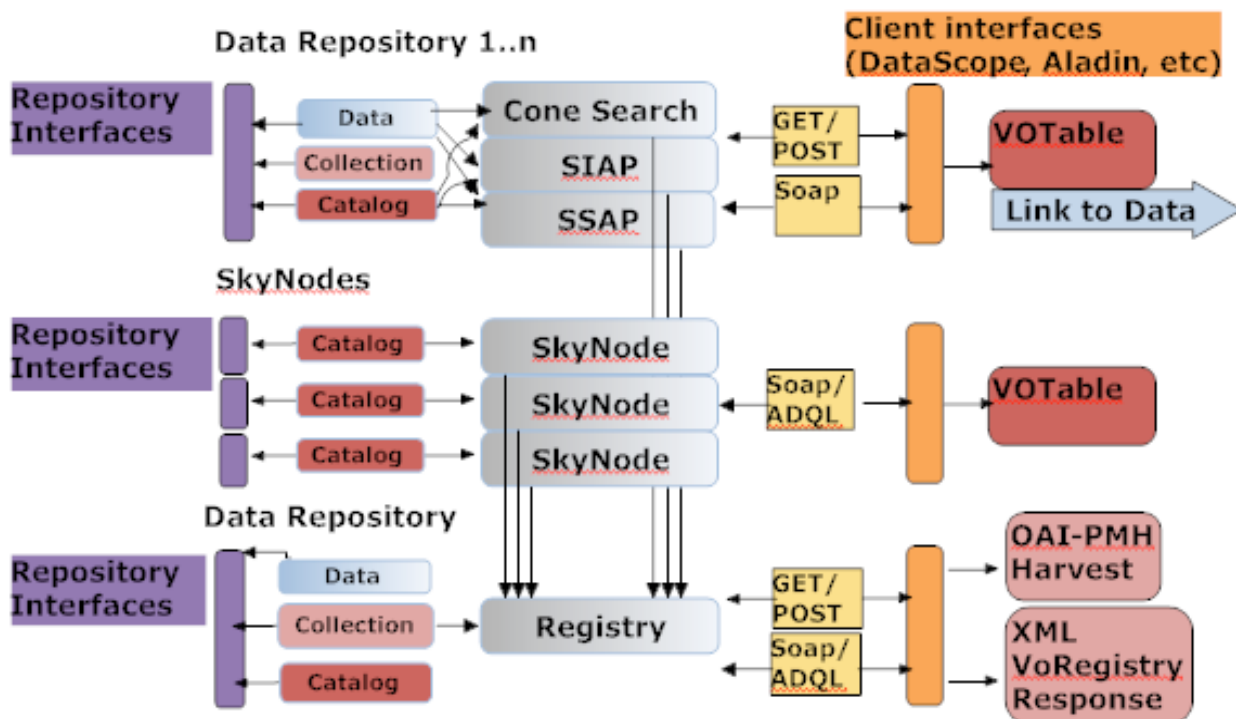


Figure 2 IVOA Architecture

Description. Most of the major worldwide astrophysics repositories are members of the alliance. The alliance members develop specifications for providing services. The specifications are often based on

capabilities that have been developed at alliance facilities. Figure 2 presents a diagram of the components of the Virtual Observatory architecture. Like the Open Web Services, the IVOA relies on data providers to implement servers for providing access to their collections. The cone search protocol can be applied to nearly any astronomical resource type so there are many cone search servers. Most of the other services apply to a specific data type or required functionality. There are a number of "Simple ... Access Protocols" protocols which provide access to images (SIAP), spectrum (SSAP) and line (SLAP) data. Each service requires position and size keywords with other special request keywords or options depending on the service type. There is one common option `FORMAT=METADATA` which will return information about the service. These services respond with a VOTable XML document, (much like an HTML table) which provides the search results as embedded tabular data and/or links to binary data files. The SkyNode service provides support for database queries in Astronomical Data Query Language (ADQL), a subset version of SQL92 with special commands and functions to support astronomical queries. The SkyNode service is used for cross matching sources across different wavelengths from different repositories. Once an IVOA service is established the data provider uses one of several registry sites to enter information about the service. The registry information uses the Dublin Core metadata and also includes curation, content, interface and capability information. Key metadata values for user searches are spectral, spatial and temporal coverage and resolution. Registry sites can be automatically harvested by other registries. There is an effort underway to develop a registry of registries.

Service Summary. Operations that deal with the registry service include:

KeywordSearch	- provides a basic text search
Search	- executes an ADQL query against the stored resources
GetRegistries	- dumps the contents of the registry.
Identify	- returns the OAI identifier for the registry
ListMetadataFormats	- lists the OAI metadata formats that the registry supports
ListSets	- lists the metadata sets that the registry supports
GetRecord	- retrieves the specified records from the registry
ListIdentifiers	- lists the identifiers of records that can be retrieved from the registry
ListRecords	- lists the records that can be retrieved from the registry

The SkyNode operations include the following:

GetAvailability	- provides service availability information
PerformQuery	- searches within a circular region
Table(s)	- provides available table names
Column(s)	- provides available column names
Formats	- identifies supported formats
Functions	- identifies supported functions
PerformQuery	- searches within a complex shape. (F)
QueryCost	- determines the object density per square degree. (F)
ExecutePlan	- performs cross matching between a survey and a VOTable. (F)
Footprint	- returns an intersection of a specified region with a survey. (F)

There are no operations defined for the "Simple" protocols in the current specification documents. All services take a position and size argument and return a VOTable.

Cone Search	- generic positional search for image, tables, catalogs, spectra, etc.
Simple Image Access	- retrieves or generates images of a selected region.
Simple Spectral Access	- retrieves 1d spectra and spectral energy distribution (SED) measurements.
Simple Line Access	- provides a search for spectral lines.

Other services that are in development include a workflow service for specifying an automated sequence of processes to be carried out; VOspace for providing and managing data storage resources within the Virtual Observatory; and VOEvent for providing a mechanism for notifying the community of the discovery of a transient celestial event. There are also a number of new simple access protocols for tables, catalogs, numerical data and a datacube extension to the SIAP protocol. There is also a set of resource services being developed including File Store Service, File Manager Service and Security Service.

The following sample urls show how to invoke a cone search server and simple image access protocol server using the HTTP GET protocol.

<a href="http://casjobs.sdss.org/vo/dr5cone/sdssConeSearch.aspx/ConeSearch?RA=195.163&amp;DEC=2.5&amp;SR=0.1">http://casjobs.sdss.org/vo/dr5cone/sdssConeSearch.aspx/ConeSearch?RA=195.163&amp;DEC=2.5&amp;SR=0.1</a> <a href="http://skyserver.sdss.org/vo/dr2siap/SIAP.aspx/getSiap?POS=195.163,2.5&amp;SIZE=0.1&amp;FORMAT=image/jpeg">http://skyserver.sdss.org/vo/dr2siap/SIAP.aspx/getSiap?POS=195.163,2.5&amp;SIZE=0.1&amp;FORMAT=image/jpeg</a>
--

Summary. The Virtual Observatory provides a bold and exciting vision for putting all astronomical resources on a user's desktop. The retrieval services can provide enough data for years worth of research in a few minutes. However, the effort required to sort out and understand those data products is still formidable. The IVOA has embraced the Service Oriented Architecture including WSDL and SOAP, though implementation will take time. There are some inconsistencies in implementations of services (differences in the metadata contents of different registries). It would seem that a set of standard server software should be available to potential data providers along with a certification program.

## **Planetary Data System - Object Oriented Data Technology**

The Planetary Data System is the principle archive system for data from NASA's Planetary Science missions. It is a federation of distributed discipline nodes which each have the responsibility for a different planetary discipline (geosciences, atmospheres, etc.). Each discipline provides storage and retrieval capabilities for its data collection. OODT is a software framework for creating, maintaining and accessing heterogeneous, distributed data resources.

Description. Figure 3 presents a diagram of the PDS-OODT architecture. OODT "profiles" provide metadata about collections of products (datasets) or individual products. Products are generally individual scientific observations or files containing bundled observations. Sometimes several related files constitute a product. The key components of the system are the three servers shown in the architecture diagram. The product server receives a target directory or file name and a return type (which specifies the packaging of the results), performs the requested action on the target, and returns the requested data in either an XML message or a data stream. The packaging options include returning size information only, returning information for a directory or directory tree, selecting associated files,

and packaging specifications (zip). In the PDS architecture the geographically distributed discipline nodes are responsible for providing access to specific categories of planetary data products (geosciences, atmospheres, rings). Each discipline node provides product servers to provide access to their products. The profile server receives a profile query and searches the product catalog to generate an XML documents identifying matching profile entries. It may also provide pointers to other profile servers which might have relevant data. The query server receives query requests and then sends the query on to applicable profile and product servers looking for matches. The query server is "seeded" with the location of references to profile servers to query. The query server collects the results from the profile servers and product servers and returns them to the requester. The Engineering Node of the PDS maintains a central query server and several profile servers that describe all the datasets that are stored at the discipline nodes. Some of the discipline nodes are developing profile servers to support metadata searches, but access to these servers is limited to PDS developed clients at this time. There is also a Catalog and Archive Service that provides a robust system for ingesting data products including data validation, metadata extraction and data manipulation processes.

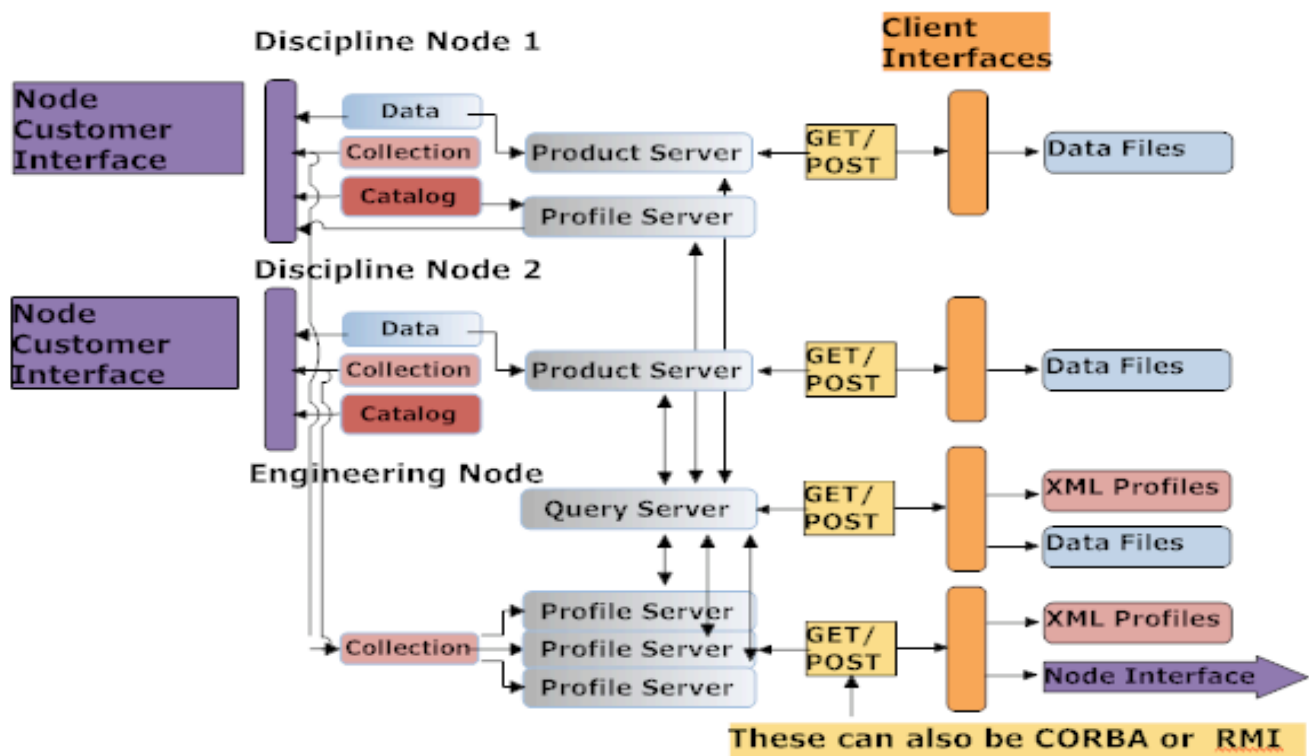


Figure 3 – PDS-ODT architecture

Services. The OODT software provides the following services:

Query Service	- provides distributed queries of profile and product servers.
Profile Service	- provides metadata descriptions of servers, datasets and individual products.
Product Service	- provides access to data products including product conversion or packaging.
Catalog and Archive Service	- provides for ingestion, validation and transformation of products.
Object Id Service	- provides unique numeric identifiers for registered objects.
Metadata Service	- registers and retrieves metadata elements, schema and profiles of resources.
Server Manager	- manages and monitors server processes on multiple platforms.

Server Controller	- controls multiple server managers.
Remote Control	- allows remote control (status, start, stop, change properties) of server managers.

A following examples show requests to a product server and a profile server using the HTTP GET protocol.

```

http://starbrite.jpl.nasa.gov/prod?object=urn:eda:rmi:PDS.Img.Product&keywordQuery=OFSN++data/mgs-m-moc-na wa-2-dsdp-l0-v1.0/mgsc 0004/sp1223/sp122304.img+AND+RT++PDS JPEG

http://starbrite.jpl.nasa.gov/q?type=profile&object=urn:eda:rmi:JPL.PDS.Profile&keywordQuery=resclass++resource+AND+dsid++ODY-M-ACCEL-5-ALTITUDE-V1.0+AND+RETURN++resourceid

```

Summary. The PDS implementation of OODT provides a standard mechanism for accessing data products stored at heterogeneous, geographically distributed discipline node sites. It is easy for a user to get a specific product, directory or entire volume of data, if the user has the explicit file, directory or volume name. The central registry of profile servers provides a limited search capability for accessing collections then the user is handed off to a discipline specific query or retrieval system with a custom catalog search. It would seem useful to have profile servers established for all discipline node catalogs so that they are more widely accessible.

**EOS Clearinghouse (ECHO).** ECHO is a centralized repository for metadata describing earth science data stored at the EOS Distributed Active Archive Centers (DAAC's) and other facilities. The metadata includes inventory (collection) and catalog (granule) descriptions as well as browse representations of the data products. The metadata is provided by Data Partners and stored in an Oracle spatial database which provides the capability for sophisticated spatial searching. The ECHO system does not provide an end-user interface to access the metadata. Rather it relies on Client Partners to develop interfaces for querying the metadata and ordering data. There is an extended service registry (in development) to identify value-added services which can be applied to data retrieved using ECHO.

Description. Figure 4 presents a diagram of the ECHO architecture. ECHO has a highly structured interface for dealing with its partner groups. Data Partners sign an agreement which specifies the support they will provide for the metadata ingestion process and the order service. The data partners map their internal metadata structure to the ECHO structure. They produce XML files (from templates supplied by ECHO) which are transferred to a repository via FTP for loading into the ECHO database. A program is available to assist users in mapping existing XML metadata to the ECHO templates by using XSLT to perform the conversion. Access to the actual data resources on the provider's system is generally via a hyperlink that is included with the metadata. Client Partners also sign an agreement which specifies the interface to the ECHO system and certain design, testing, reporting and client customer support requirements. All functionality is accessed through a set of application program interfaces (API's) which are transmitted in XML enclosed in SOAP messages. The primary role of the client partners is to develop user interfaces which generate queries to the ECHO Catalog Service and orders to the Order Entry Service. Queries are categorized as a "discovery search" on collections (datasets), or an "inventory search" on granules (product catalogs). Because of the massive size of the database, it is very important that queries are constructed optimally. The data granules that are identified in a query can then be added to an order that is transmitted to a data provider for execution and delivery. Service partners are allowed to advertise value-added services (special processing, data



mining) that operate on the data that can be retrieved via ECHO. They agree to maintain their services and to provide WSDL service definitions which are entered in the UDDI service registry.

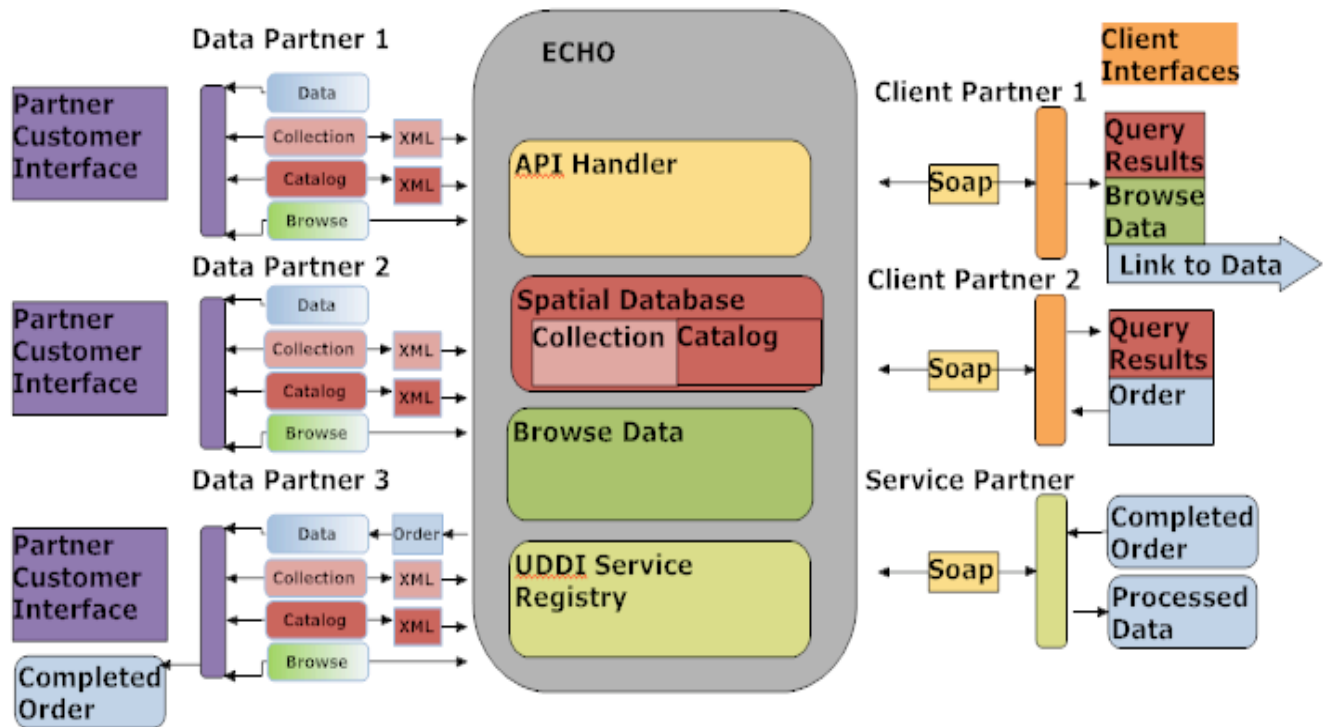


Figure 4 - Echo Architecture

Service summary. ECHO services are invoked via the Session Manager interface. Four operations are supported which include: "identify" to specify the client; "login" to establish a user session; "perform" to submit XML messages requesting services; and "logout" to terminate the user session. The interface to these services is an XML document containing the service name, the operation name and the required parameters for the operation. Each service has a large number of operations, too many to go through in detail. The core retrieval services include:

Catalog Service	- submit and manage queries and results
Order Entry Service	- asynchronous service that allows the user to construct an order
Provider Profile Service	- lists provider information
Subscription Service	- selectively receive metadata updates when providers add metadata
Extended Service Management Service - create and maintain extended service interfaces.	

The key Catalog Service operation is ExecuteQuery which executes a query and generates a result set. Other operations include SaveQuery, SaveResultSets, GetQueryResults, ExecuteSavedQuery. Order Entry Service operations include CreateOrder, AddOrderItems, QuoteOrder, ValidateOrder, SubmitOrder and CancelOrder.

ECHO also has a number of administrative services which include:

Administration Service	- provide internal administrative functions.
------------------------	--

Data Management Service	- control access to data by providers.
Group Management Service	- user group management by providers.
Provider Account Service	- set policy and maintain notification subscriptions.
Provider Order Management Service	- submit, quote or cancel user orders.
Registration Service	- request a new user account.
User Account Service	- provides user account maintenance functions.

Summary. ECHO is committed to using the emerging "Web Services" architecture (SOAP, WSDL, UDDI). The ECHO model of a centralized metadata catalog contrasts with the other architectures described in this paper. Two of the main features that are cited are the use of the Oracle spatial data base for sophisticated geographic searches and the fact that if a data provider system is not available ECHO will still have their catalog on-line. In this architecture the data providers could conceivably forgo any investment in user interface, security and other administrative issues and leave it all up to ECHO and the Client Partners. The ECHO system seems to always be in transition from one version to the next and some documentation refers to past versions and some to future versions, which can be confusing.

### Access Architecture Comparisons.

The four architectures are presented in order of the degree of centralization. While the specification process for OWS is centralized the actual implementations are completely decentralized. Registries seem to be maintained by fairly narrow user groups and it takes quite a bit of effort to locate WMS servers. There is a bit more order to the IVOA partly because most of the service providers are large facilities with many data collections to offer. Generally the facilities also provide a registry for locating resources and retrieval systems that have been developed to access IVOA resources (DataScope, for example). The PDS-OODT system provides a limited central registry and search interface that points to the search resources (customized for specific mission collections) and product servers at the discipline nodes. Processing services (mostly format conversion and packaging) are provided at the distributed repositories with standard software. The ECHO system is highly centralized with all searchable metadata in the central database. In the ECHO system any processing services are provided by the distributed repositories or by extended service providers.

	OWS	VOA	PDS-OODT	ECHO
"Web Services" Protocols	WSDL, SOAP (future)	WSDL, SOAP	NONE	WSDL, SOAP, UDDI
Query Languages	CQL FILTER	ADQL/VOQL	OODT-XMLQUERY	IMSACL
Message protocols	HTTP GET/ POST, SOAP	HTTP GET/POST, SOAP	HTTP GET/POST, RMI, CORBA	XML, SOAP
Primary Data Format	GEOTIFF	FITS	PDS	HDF
Browse Formats	JPEG	JPEG	JPEG	HDF
Capabilities Request	REQUEST=GetCapabilities	FORMAT=METADATA	NONE (future)	NONE
Text Response format	XML document	VOTable XML Document	XML Profiles	XML document in SOAP
Data Access	Stream	Stream or Link	Stream	Link
Registry Protocol	ebRIM or ISO19119	OAI-PMH	Custom	UDDI
Registry Architecture	Distributed with Harvest	Distributed, Harvestible	Distributed with links	Centralized no Harvest
Registry Harvesting	From Servers	From Registries, not servers	From data volumes	NONE
Registry Metadata	Dublin Core Subset	Dublin Core +	Dublin Core +	Custom
Product Metadata	ISO19115,Z39.50 GEO/C	Space Time Coord + FITS	PDS Data Dictionary	EOSDIS V0 ISO19115*

Table 1 Feature Comparison

Messaging protocols. OWS supports HTTP/GET-POST and HTTP/SOAP for interacting with WMS, WFS and WCS servers. It supports these protocols as well as CORBA and Z39.50 for interacting with the CSW. IVOA supports HTTP/GET-POST (the vast majority) and HTTP/SOAP for cone search and S\*AP servers. It supports HTTP/SOAP for SkyNode and registry servers. It also supports OAI-PMH for registry harvesting. PDS-ODT supports HTTP/GET-POST, CORBA or RMI protocols for communicating with and between servers. ECHO supports FTP for transferring catalog data (XML files) and browse data to the central site. Interaction with the ECHO APIs is via HTTP/SOAP.

Data Formats. Each of these systems is oriented toward the common data formats used within the discipline. These are GEOTIFF and GML for OWS, FITS for astronomy, PDS format for Planetary and HDF for ECHO. OWS, IVOA and PDS are all able to translate their internal formats into JPEG and some support is also provided for GIF or PNG formats. ECHO is limited to HDF format for browse images.

Capabilities Request. The OWS services all provide a GetCapabilities option. Other operations successively provide more detail (categories of information that are available, then schemas for that information, then data to load into schemas). The IVOA has a similar but not as robust a capability using the FORMAT=METADATA option on S\*AP services. PDS-ODT is planning a similar feature as part of the Planetary Data Access Protocol (PDAP) specification.

Text Response formats. All four architectures provide responses to most operations in the form of XML text files. The VOTable produced by nearly all IVOA services deserves special attention. It includes one or more "resource" elements with embedded "table" or "resource" elements. Descriptive elements can appear at the root level or any level below that to indicate their applicability to embedded tables. A series of "field" statements (with attributes name, id, datatype, etc.) describe each column of the table. The table data is enclosed in a "data" tag. Normally the output uses the "tabledata" serialization option, where the table is embedded in XML using "tr" and "td" tags to delimit rows and columns. Other delivery options include "FITS" serialization and "binary" serialization. The table can be formatted for viewing with an extensible stylesheet language transformations (XSLT) processor or viewed directly with an XSLT aware browser. It can also be used as a source list for a number of image and spectral display and analysis programs. The PDS-ODT system has incorporated the VOTable as an output option in its Planetary Data Access Protocol (PDAP) specification.

Data Access. In normal operations the OWS and PDS-ODT servers stream data products back to the user. IVOA can embed and transmit binary data in a VOTable or can provide hyperlinks to the data products. ECHO provides links to the data which is stored at the data partner sites. ECHO also provides an integrated order capability which is carried out by the data partners. There is the capability in IVOA for asynchronous data retrieval activities, but no specific order capability. PDS-ODT has a separate order system that is not integrated with the OODT software.

Security. For the most part these systems provide public access to all data so security is not a major concern. The only provision for security in OWS is through the session capability of the CSW service. There are add-on security resources available in IVOA and OODT. ECHO users log on with a userid

and password, though a guest account can be used to access the search services. ECHO has a built-in authentication service.

**Documentation.** It is extremely difficult to find documentation that clearly describes the operation of any of these architectures. Most of the documentation for OWS and IVOA is in the form of specification documents. The IVOA summer school website provides papers and presentations describing the technologies that IVOA is using and the application of the services for doing science research. The OODT web site provides papers and presentations describing the PDS implementation of OODT as well as software documentation and a rudimentary user guide. The ECHO site provides a user guide, API guide and workshop presentations.

**Metadata.** All the systems except ECHO use the Dublin Core for collection metadata. Within OWS two application profiles have been developed for the catalog specification, one which uses the ISO-19115 (geographic metadata) and ISO 19119 (service metadata) standards and one which uses the OASIS ebXML Registry Information Model (ebRIM). There is also a Z39.50 protocol binding for the catalog specification which specifies the use of the geospatial metadata (GEO) and catalog interoperability protocol (CIP) profiles. The Universal Description, Discovery and Integration (UDDI) service registry was evaluated as a catalog server, but the architecture wasn't a good fit for supporting the metadata query requirements. Within IVOA a set of metadata values is specified for the registry including collection and service content metadata (instrument, coverage, resolution); data quality metadata and service metadata. Since most data products are in FITS format, the metadata most often encountered with products is the set of standard FITS keywords. The Unified Content Descriptors (UCD) specification provides a controlled vocabulary of logically constructed identifiers to define the exact type of quantity (e.g. pos.eq.dec for declination in equatorial coordinates). These identifiers do not provide the element name or units. The Space Time Coordinate (STC) specification provides coordinate metadata for a resource profile, search location, catalog entry location or observation location plus observatory location elements. The VOEvent specification defines a standard information packet for representing, transmitting, publishing and archiving the discovery of a transient celestial event. Within PDS-OODT there are a few special elements for profile metadata but all product metadata elements are based on names defined in the Planetary Data System Data Dictionary, a strictly controlled vocabulary of planetary science terms. For ECHO the metadata model is derived directly from that used by the Earth Observing System Data and Information System (EOSDIS) Core System (ECS) and conforms to the Federal Geographic Data Committee (FGDC) and Global Change Master Directory (GCMD) standards. The schema can be extended with product specific metadata. The Extended Services capability uses a UDDI registry.

**Software.** The OWS is supported by several free server and client implementations which support Windows, Linux and Macintosh platforms, among them MapServer, GeoServer, Deegree and UDIG (client only). Most of the major GIS vendors support at least the WMS specification in their servers. The WMS interface is simple enough that most implementations are developed by the data provider. Virtually all astrophysics web sites and many tools (Aladin, OASIS) make use of IVOA services, but there are no off-the-shelf packages for establishing a service interface. There is a repository of sample software for providing service implementations available through the National Virtual Observatory. Other resources are scattered around at the IVOA sites. OODT provides a large library of Java software for creating, maintaining and populating product and profile servers. New implementations using OODT (applying the catalog and archive system to a new project for example) require very little code

development. The ECHO server software is available for download. Tools for preparing and submitting metadata to the system and sample client software implementations are available.

Usage. According to the OPENGIS web page there are on the order of 150 map, 90 feature, 20 coverage and 10 catalog service implementations supporting various versions of the implementation specifications. More than 318 vendors and institutions are identified on the OPENGIS web site. For IVOA, the total number of registered services as of mid-2006 is: Cone Search (244), Simple Image Access (85), Simple Spectrum Access (10), Open Sky Node services (25), Tabular Sky Services (11643), Sky Services (8) plus a few dozen other services. The OODT software has been applied to about a dozen space science, earth science, computer modeling and medical applications. OODT product servers are in operation at all the PDS discipline node sites and are used to provide access to most on-line PDS data. The ECHO system is by its nature unique. There are eleven data providers. The client partner page shows one operational client and five in development. I have been unable to determine whether there are any service partners at this time.

Certification and quality control. The OGC has a certification and testing program to assure that implementations meet its specifications. It might be useful for IVOA and PDS-OODT to have a similar capability. The ECHO system requires testing of client partners interfaces.

Service Summary. Table 2 provides a summary of services for each architecture. Considering that each architecture has similar goals it is interesting to see that the list of services is fairly divergent. An attempt to align the services across systems is frustrating. The only observation I can make is that OWS and IVOA organize services by data type. PDS-OODT and ECHO embed the data type information in the server metadata and present more generic services. Over all, it would seem that the determination of what are the proper services for a information system is a complex effort.

OWS	VOA	PDS-OODT	ECHO
Catalog Service for Web	Registry Service	Query Service	Catalog Service
Web Map Service	Sky Node	Profile Service	Order Management Service
Web Feature Service	Cone Search	Product Service	Order Processing Service
Web Coverage Service	Simple Image Access	Catalog and Archive Service	Taxonomy Service
	Simple Spectrum Access	Metadata Service	Data Management Service
	Simple Line Access	Object ID Service	Extended Services Service
	Sky Service	Server Manager	Group Management Service
	Tabular Sky Service	<i>Order Service*</i>	Invocation Service
	File Manager Service		Invocation Utility Service
	File Store Service		Provider Service
	Policy Manager Service		Status Service
	Security Service	<i>Notification Service*</i>	Subscription Service
	Security Manager		Authentication Service
			User Service
		* not integrated in OODT	Administration Service

Table 2 Service Summary

## Exemplary or Noteworthy Features

The following list identifies exemplary or noteworthy features of the various architectures.

1. The layered discovery model (for example, GetCapabilities, DescribeRecords, GetRecords) of the OWS scheme.
2. The ability of the OWS registry to harvest from servers. Other schemes require some manual efforts for registry creation. The IVOA registries can be harvested and can harvest from each other, but are fed manually.
3. The multi-talented VOTable. Having a standard output from many different services seems to make it easier to create great tools.
4. The cone search to access virtually every astronomical resource for a patch of the sky.
5. The use of web service protocols by IVOA, ECHO and to a lesser extent OWS. There are many web services skeptics around, but within OWS, IVOA, and ECHO there are also many Web Services proponents. The most common stated benefit is cross platform compatibility. ECHO cites the use of an industry standard protocol, a simplified mechanism to connect applications, an interface that is self-describing and allows automated discovery, an interface that is supported by a large number of vendors and publicly available development tools. Michael Burnett of ECHO says the use of web services radically cuts development time for building new interfaces to the catalog ("guettes"). IVOA documentation cites self-description through the WSDL file, a true client API, an exception mechanism, routing through gateways, and a security model as advantages of SOAP services.
6. The UNSTANDARDIZATION of XML-based query languages. Each system uses its own XML query syntax and thus requires special programs to generate that syntax from the user's input.
7. The OWS and IVOA combine the service metadata and collection metadata in a single registry. It seems like these are quite different types of information and that separate registries might be a better idea.
8. The IVOA holds an excellent summer studies program each year to familiarize astronomers with the emerging technologies and the IVOA services and data.

## References

[Open Geospatial \(OWS\) Website](#) (specifications)  
[International Virtual Observatory Alliance \(IVOA\) Document Website](#) (specifications)  
[Object Oriented Data Technology \(OODT\) Website](#) (papers, software, documentation)  
[EOS Clearinghouse \(ECHO\) Website](#) (documents and specifications)  
[Virtual Observatory Architecture Overview](#), 2004-06-14, Roy Williams, et al.  
[Web Services Architecture](#), W3C Working Group Note 11 February 2004  
[Migrating to a service-oriented architecture, Part 1](#), 16 Dec 2003, IBM  
[Architectural Styles and the Design of Network-based Software Architectures](#), 2000, Roy Thomas Fielding